C 13

```
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSSSSS
RRR         RRR  MMMMMM   MMMMMM  SSS
RRR         RRR  MMMMMM   MMMMMM  SSS
RRR         RRR  MMMMMM   MMMMMM  SSS
RRR         RRR  MMM   MMM   MMM  SSS
RRR         RRR  MMM   MMM   MMM  SSS
RRR         RRR  MMM   MMM   MMM  SSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSS
RRR   RRR        MMM         MMM              SSS
RRR   RRR        MMM         MMM              SSS
RRR   RRR        MMM         MMM              SSS
RRR      RRR     MMM         MMM              SSS
RRR      RRR     MMM         MMM              SSS
RRR      RRR     MMM         MMM              SSS
RRR         RRR  MMM         MMM  SSSSSSSSSSSS
RRR         RRR  MMM         MMM  SSSSSSSSSSSS
RRR         RRR  MMM         MMM  SSSSSSSSSSSS
```

E 13

```
**FILE**ID**RM3POSRFA
```

```
RRRRRRRR    MM      MM    333333    PPPPPPPP      000000      SSSSSSSS  RRRRRRRR    FFFFFFFFFF    AAAAAA
RRRRRRRR    MM      MM    333333    PPPPPPPP      000000      SSSSSSSS  RRRRRRRR    FFFFFFFFFF    AAAAAA
RR      RR  MMMM  MMMM  33      33  PP      PP  00      00  SS          RR      RR  FF          AA      AA
RR      RR  MMMM  MMMM  33      33  PP      PP  00      00  SS          RR      RR  FF          AA      AA
RR      RR  MM  MM  MM          33  PP      PP  00      00  SS          RR      RR  FF          AA      AA
RR      RR  MM  MM  MM          33  PP      PP  00      00  SS          RR      RR  FF          AA      AA
RRRRRRRR    MM      MM        33    PPPPPPPP    00      00    SSSSSS    RRRRRRRR    FFFFFFFF    AA      AA
RRRRRRRR    MM      MM        33    PPPPPPPP    00      00    SSSSSS    RRRRRRRR    FFFFFFFF    AA      AA
RR  RR      MM      MM          33  PP          00      00          SS  RR  RR      FF          AAAAAAAAAA
RR  RR      MM      MM          33  PP          00      00          SS  RR  RR      FF          AAAAAAAAAA
RR    RR    MM      MM  33      33  PP          00      00          SS  RR    RR    FF          AA      AA      ....
RR    RR    MM      MM  33      33  PP          00      00          SS  RR    RR    FF          AA      AA      ....
RR      RR  MM      MM    333333    PP            000000    SSSSSSSS    RR      RR  FF          AA      AA      ....
RR      RR  MM      MM    333333    PP            000000    SSSSSSSS    RR      RR  FF          AA      AA      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

RM3
VO4

```
;     1          0001  0 MODULE RM3POSRFA (LANGUAGE (BLISS32) ,
;     2          0002  0                 IDENT = 'V04-000'
;     3          0003  0                 ) =
;     4          0004  1 BEGIN
;     5          0005  1 !
;     6          0006  1 !****************************************************************************
;     7          0007  1 !*                                                                          *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                     *
;    11          0011  1 !*                                                                          *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    17          0017  1 !*  TRANSFERRED.                                                            *
;    18          0018  1 !*                                                                          *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    21          0021  1 !*  CORPORATION.                                                            *
;    22          0022  1 !*                                                                          *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
;    25          0025  1 !*                                                                          *
;    26          0026  1 !*                                                                          *
;    27          0027  1 !****************************************************************************
;    28          0028  1
;    29          0029  1 !++
;    30          0030  1 !
;    31          0031  1 ! FACILITY:      RMS32 index sequential file organization
;    32          0032  1 !
;    33          0033  1 ! ABSTRACT:
;    34          0034  1 !                This module positions to a record by RFA.
;    35          0035  1 !
;    36          0036  1 !
;    37          0037  1 ! ENVIRONMENT:
;    38          0038  1 !
;    39          0039  1 !               VAX/VMS Operating System
;    40          0040  1 !
;    41          0041  1 !--
;    42          0042  1
;    43          0043  1 !
;    44          0044  1 ! AUTHOR:        Todd M. Katz            CREATION DATE:          14-Jan-83
;    45          0045  1 !
;    46          0046  1 !
;    47          0047  1 ! MODIFIED BY:
;    48          0048  1 !
;    49          0049  1 !       V03-003 TSK0001         Tamar Krichevsky        15-Jun-1983
;    50          0050  1 !               Change addressing mode for RM$RU_RECLAIM to log relative.
;    51          0051  1 !
;    52          0052  1 !       V03-002 TMK0001         Todd M. Katz            11-Mar-1983
;    53          0053  1 !               If it possible that some reclamation will be required (the
;    54          0054  1 !               file is write accessed and RU Jouranallable), then make sure
;    55          0055  1 !               the primary data bucket is accessed exclusively.
;    56          0056  1 !
;    57          0057  1 !       V03-001 MCN0001         Maria del C. Nasr       24-Feb-1983
;   58         0058  1 !                         Reorganize Linkages
;   59         0059  1 !
;   60         0060  1 !
;   61         0061  1 !*****
;   62         0062  1
;   63         0063  1 LIBRARY 'RMSLIB:RMS';
;   64         0064  1
;   65         0065  1 REQUIRE 'RMSSRC:RMSIDXDEF';
;   66         0130  1
;   67         0131  1 ! Define default PSECTS for code.
;   68         0132  1 !
;   69         0133  1 PSECT
;   70         0134  1     CODE = RM$RMS3(PSECT_ATTR),
;   71         0135  1     PLIT = RM$RMS3(PSECT_ATTR);
;   72         0136  1
;   73         0137  1 ! Linkages.
;   74         0138  1 !
;   75         0139  1 LINKAGE
;   76         0140  1     L_RABREG_67,
;   77         0141  1     L_PRESERVE1;
;   78         0142  1
;   79         0143  1 ! External Routines.
;   80         0144  1 !
;   81         0145  1 EXTERNAL ROUTINE
;   82         0146  1     RM$FIND_BY_RRV      : RL$RABREG_67,
;   83         0147  1     RM$RLSBKT           : RL$PRESERVE1,
;   84         0148  1     RM$RU_RECLAIM       : RL$RABREG_67 ADDRESSING_MODE( LONG_RELATIVE );
;   85         0149  1
```

```
1 GLOBAL ROUTINE RM$POS_RFA : RL$RABREG_67 =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine positions to a primary data record by its RFA.
1 !
1 !       If RMS finds that the record with the appropriate RFA is marked
1 !       RU_DELETE and the Recovery Unit in which the record was deleted is
1 !       still active, then RMS returns positioned to this record and lets a
1 !       higher level routine decide what to do. If the Recovery Unit in which
1 !       the record was deleted has successfully terminated, then RMS will
1 !       return an error of DEL after deleting this RU_DELETEd record (if it has
1 !       write access to the file), and releasing the primary data bucket.
1 !
1 !       If RMS encounters a record that is marked RU_UPDATE and is in a special
1 !       format then RMS will return positioned to this record after
1 !       reformatting it. The reformatting is done if RMS has write access to
1 !       the file, and the Recovery Unit in which it was updated has
1 !       successfully terminated.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       RM$POS_RFA()
1 !
1 ! INPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       IFAB                    - address of IFAB
1 !           IFB$V_RU            - if set, the file is Recovery Unit Journallable
1 !           IFB$V_WRTACC        - if set, file is open for write access
1 !
1 !       RAB                     - address of RAB
1 !           RAB$L_RFA0          - RFA VBN of primary data record to be retrieved
1 !           RAB$W_RFA4          - RFA IF of primary data record to be retrieved
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       IRAB                    - address of IRAB
1 !           IRB$L_CURBDB        - address of BDB for primary data bucket
1 !           IRB$W_RFA_ID        - ID of current primary data record
1 !           IRB$L_RFA_VBN       - VBN of current primary data bucket
1 !
1 !       REC_ADDR                - address of primary data record
1 !
1 ! ROUTINE VALUE:
1 !
1 !       SUC     - the primary data record with this RFA has been found.
1 !       DEL     - the primary data record with this RFA has been deleted.
1 !       RNF     - the primary data record with this RFA has not been found.
1 !
```

```
1 ! SIDE EFFECTS:
1 !
1 !       On success, REC_ADDR points to the non-deleted primary data record
1 !           and the BDB of the primary data bucket maybe found in IRB$L_CURBDB.
1 !       On failures, all accessed buckets are released.
1 !       If the record is marked RU_DELETEd, it might have been deleted.
1 !       If the record is marked RU_UPDATEd, it might have been reformatted.
1 !
1 !--
1
2     BEGIN
2
2     BUILTIN
2         AP;
2
2     EXTERNAL REGISTER
2         COMMON_RAB_STR,
2         R_IDX_DFN_STR,
2         R_REC_ADDR_STR;
2
2     LOCAL
2         STATUS;
2
2     ! If it possible that some reclamation will be required (the file is write
2     ! accessed and RU Jouranallable), then make sure the primary data bucket is
2     ! accessed exclusively.
2     !
2     IF  .IFAB[IFB$V_WRTACC]
2        AND
2       .IFAB[IFB$V_RU]
2     THEN
2         IRAB[IRB$B_CACHEFLGS] = CSH$M_LOCK;
2
2     ! RMS is successful at positioning to the primary data record by its RFA.
2     !
2     IF (STATUS = RM$FIND_BY_RRV (.RAB[RAB$L_RFA0], .RAB[RAB$W_RFA4], 0))
2     THEN
3         BEGIN
3
3         ! If RMS finds that the record with this RFA has been deleted within a
3         ! Recovery Unit, then it subjects this record to further processing
3         ! before deciding whether to return this record, or to return an error
3         ! of record deleted.
3         !
3         IF .REC_ADDR[IRC$V_RU_DELETE]
3         THEN
4             BEGIN
4
4             LOCAL
4                 TEMP_STATUS;
4
4             ! If the Recovery Unit in which the record was deleted has
4             ! not completed or if the file has not been opened for write
4             ! access then no space can be reclaimed. Return an error of
4             ! RMS$_DEL unless the record itself could not be locked in which
4             ! case return the record so that caller may decide what to do with
4             ! it.
```

```
;  201         0264  4                    !
;  202         0265  5                    IF NOT (TEMP_STATUS = RM$RU_RECLAIM())
;  203         0266  4                    THEN
;  204         0267  5                        BEGIN
;  205         0268  5
;  206         0269  6                        IF .TEMP_STATUS<0,16> NEQU RMSERR(RLK)
;  207         0270  5                        THEN
;  208         0271  5                            STATUS = RMSERR(DEL);
;  209         0272  5                        END
;  210         0273  5
;  211         0274  5                    ! If the Recovery Unit in which the record was deleted has
;  212         0275  5                    ! successfully terminated, then RMS will be able to delete the
;  213         0276  5                    ! record at this time. In this case, change the return status
;  214         0277  5                    ! to RMS$_DEL.
;  215         0278  5                    !
;  216         0279  4                    ELSE
;  217         0280  4                        STATUS = RMSERR(DEL);
;  218         0281  4                    END
;  219         0282  4
;  220         0283  4                ! If the primary data record RMS has positioned was updated within a
;  221         0284  4                ! Recovery Unit, then re-format the record at this time provided
;  222         0285  4                ! the file is opened for write access, and the Recovery Unit has
;  223         0286  4                ! terminated successfully.
;  224         0287  4                !
;  225         0288  3                ELSE
;  226         0289  3                    IF   .REC_ADDR[IRC$V_RU_UPDATE]
;  227         0290  3                      AND
;  228         0291  3                        .IFAB[IFB$V_WRTACC]
;  229         0292  3                    THEN
;  230         0293  3                        RM$RU_RECLAIM();
;  231         0294  3
;  232         0295  3                ! If the record originally positioned to has been found to be deleted,
;  233         0296  3                ! then release the primary data bucket in which it was found.
;  234         0297  3                !
;  235         0298  4                IF .STATUS<0,16> EQLU RMSERR(DEL)
;  236         0299  3                THEN
;  237         0300  4                    BEGIN
;  238         0301  4
;  239         0302  4                    GLOBAL REGISTER
;  240         0303  4                        R_BDB_STR;
;  241         0304  4
;  242         0305  5                    RELEASE (IRAB[IRB$L_CURBDB])
;  243         0306  4                    END
;  244         0307  4
;  245         0308  4                ! Otherwise, setup to update the NRP context with the current address
;  246         0309  4                ! in the file of this primary data record.
;  247         0310  4                !
;  248         0311  3                ELSE
;  249         0312  4                    BEGIN
;  250         0313  4                    IRAB[IRB$L_RFA_VBN] = .BBLOCK[.IRAB[IRB$L_CURBDB], BDB$L_VBN];
;  251         0314  4                    IRAB[IRB$W_RFA_ID]  = IRC$_ID(REC_ADDR);
;  252         0315  3                    END;
;  253         0316  3                END
;  254         0317  3
;  255         0318  3            ! RMS was not successful at positioning to the primary data record by its
;  256         0319  3            ! RFA. Setup to return the reason why.
;  257         0320  3            !
;   258         0321  2    ELSE
;   259         0322  3        BEGIN
;   260         0323  3        IRAB[IRB$L_CURBDB] = 0;
;   261         0324  3
;   262         0325  4        IF .STATUS<0,16> EQLU RMSERR(EOF)
;   263         0326  3        THEN
;   264         0327  3            STATUS = RMSERR(RNF);
;   265         0328  2        END;
;   266         0329  2
;   267         0330  2    ! Return status.
;   268         0331  2    !
;   269         0332  2    RETURN .STATUS;
;   270         0333  1    END;



                                                                        .TITLE  RM3POSRFA
                                                                        .IDENT  \V04-000\

                                                                        .EXTRN  RM$FIND_BY_RRV, RM$RLSBKT
                                                                        .EXTRN  RM$RU_RECLAIM

                                                                        .PSECT  RM$RMS3,NOWRT,  GBL,  PIC,2

                                                     14 BB 00000 RM$POS_RFA::
                                                                        PUSHR   #^M<R2,R4>                                 ; 0150
                                      0A        06   AA E9 00002        BLBC    6(IFAB), 1$                                ; 0234
                         04    00A0   CA             01 E1 00006        BBC     #1, 160(IFAB), 1$                          ; 0236
                                 40   A9             01 90 0000C        MOVB    #1, 64(IRAB)                               ; 0238
                                                     7E D4 00010 1$:    CLRL    -(SP)                                      ; 0242
                                      7E        14   A8 3C 00012        MOVZWL  20(RAB), -(SP)                             ;
                                                10   A8 DD 00016        PUSHL   16(RAB)                                    ;
                                                   0000G 30 00019       BSBW    RM$FIND_BY_RRV                             ;
                                      5E             0C C0 0001C        ADDL2   #12, SP                                    ;
                                      52             50 D0 0001F        MOVL    R0, STATUS                                 ;
                                      61             52 E9 00022        BLBC    STATUS, 8$                                 ;
                         17           66             05 E1 00025        BBC     #5, (REC_ADDR), 3$                         ; 0251
                                         00000000G   EF 16 00029        JSB     RM$RU_RECLAIM                              ; 0265
                                      07             50 E8 0002F        BLBS    TEMP_STATUS, 2$                            ;
                               82AA   8F             50 B1 00032        CMPW    TEMP_STATUS, #33450                        ; 0269
                                                     15 13 00037        BEQL    4$                                         ;
                                      52      8262   8F 3C 00039 2$:    MOVZWL  #33378, STATUS                             ; 0280
                                                     0E 11 0003E        BRB     4$                                         ; 0251
                         0A           66             06 E1 00040 3$:    BBC     #6, (REC_ADDR), 4$                         ; 0289
                                      06        06   AA E9 00044        BLBC    6(IFAB), 4$                                ; 0291
                                         00000000G   EF 16 00048        JSB     RM$RU_RECLAIM                              ; 0293
                               8262   8F             52 B1 0004E 4$:    CMPW    STATUS, #33378                             ; 0298
                                                     11 12 00053        BNEQ    5$                                         ;
                                      54        20   A9 D0 00055        MOVL    32(IRAB), BDB                              ; 0305
                                                20   A9 D4 00059        CLRL    32(IRAB)                                   ;
                                                     7E D4 0005C        CLRL    -(SP)                                      ;
                                                   0000G 30 0005E       BSBW    RM$RLSBKT                                  ;
                                      5E             04 C0 00061        ADDL2   #4, SP                                     ;
                                                     2F 11 00064        BRB     9$                                         ; 0298
                                      50        20   A9 D0 00066 5$:    MOVL    32(IRAB), R0                               ; 0313
                                 70   A9        1C   A0 D0 0006A        MOVL    28(R0), 112(IRAB)                          ;
                                      03      00B7   CA 91 0006F        CMPB    183(IFAB), #3                              ; 0314
                                                     06 1E 00074        BGEQU   6$                                         ;
```

```
                     50       01    A6  9A 00076        MOVZBL  1(REC_ADDR), R0                               ;
                                    04  11 0007A        BRB     7$                                            ;
                     50       01    A6  3C 0007C 6$:    MOVZWL  1(REC_ADDR), R0                               ;
                74   A9             50  B0 00080 7$:    MOVW    R0, 1T6(IRAB)                                 ;
                                    0F  11 00084        BRB     9$                                            ; 0242
                              20    A9  D4 00086 8$:    CLRL    32(IRAB)                                      ; 0323
              827A   8F             52  B1 00089        CMPW    STATUS, #33402                                ; 0325
                                    05  12 0008E        BNEQ    9$                                            ;
                     52     82B2    8F  3C 00090        MOVZWL  #33458, STATUS                                ; 0327
                     50             52  D0 00095 9$:    MOVL    STATUS, R0                                    ; 0332
                                    14  BA 00098        POPR    #^M<R2,R4>                                    ; 0333
                                    05  0009A           RSB                                                   ;

; Routine Size:  155 bytes,    Routine Base:  RM$RMS3 + 0000


;  271          0334  1
;  272          0335  1 END
;  273          0336  1
;  274          0337  0 ELUDOM




;                             PSECT SUMMARY
;
;       Name                   Bytes                       Attributes
;
;    RM$RMS3                       155  NOVEC,NOWRT, RD , EXE,NOSHR, GBL, REL, CON, PIC,ALIGN(2)




;                          Library Statistics
;
;                                        -------- Symbols --------       Pages        Processing
;       File                             Total    Loaded   Percent       Mapped       Time
;
;    _$255$DUA28:[RMS.OBJ]RMS.L32;1       3109       43         1         154          00:00.4





;                             COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:RM3POSRFA/OBJ=OBJ$:RM3POSRFA MSRC$:RM3POSRFA/UPDATE=(ENH$:RM3POSRFA)

; Size:          155 code + 0 data bytes
; Run Time:         00:05.2
; Elapsed Time:     00:16.5
; Lines/CPU Min:     3866
; Lexemes/CPU-Min: 14627
; Memory Used:  72 pages
; Compilation Complete
```

0326 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY